AF$

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.: 09/752,654
Filed: December 27, 2000
Inventor(s):
Justin Chickles
Raghavendra Gurur

Title: GRAPHICAL USER INTERFACE INCLUDING PALETTE WINDOWS WITH AN IMPROVED NAVIGATION INTERFACE

§
§
§
§
§
§
§
§
§
§
§
§
§
§

Examiner: Vu, Kieu D.
Group/Art Unit: 2173
Atty. Dkt. No: 5150-43100

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to Commissioner for Patents, Alexandria, VA 22313-1450, on the date indicated below.

Jeffrey C. Hood

Signature 5/20/2005
Date

APPEAL BRIEF

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir/Madam:

Further to the Notice of Appeal filed April 14, 2005, Appellant presents this Appeal Brief. Appellant respectfully requests that this appeal be considered by the Board of Patent Appeals and Interferences.

I. REAL PARTY IN INTEREST

The subject application is owned by National Instruments Corporation, a corporation organized and existing under and by virtue of the laws of the State of Delaware, and having its principal place of business at 11500 N. MoPac Expressway, Bldg. B, Austin, Texas 78759, as evidenced by the assignment recorded at Reel 011428, Frame 0347.

II. RELATED APPEALS AND INTERFERENCES

No other appeals, interferences or judicial proceedings are known which would be related to, directly affect or be directly affected by or have a bearing on the Board's decision in this Appeal.

III. STATUS OF CLAIMS

Claim 73-76, 78-120, 122-144, and 146-158 are pending. Claims 73-76, 78-120, 122-144, and 146-158 are rejected and are the subject of this Appeal Brief. A copy of claims 73-76, 78-120, 122-144, and 146-158 as on appeal is included in the Claims Appendix attached hereto.

IV. STATUS OF AMENDMEMNTS

No amendments to the claims have been submitted subsequent to the final rejection.

V. SUMMARY OF CLAIMED SUBJECT MATTER

Independent claim 73 is directed to a computer-implemented method for navigating among a hierarchy of palette windows in a graphical user interface displayed on a computer system, where one or more of the palette windows include user-selectable palette items for including functionality in a program being created or modified. *See, e.g.,* Specification p. 5, lines 6-9. A first palette window from the hierarchy of palette windows may be displayed on a display. *See, e.g.,* Specification p. 23, lines 15-20; p. 28, lines 3-7; Figures 5A-5E at 200; Figures 6A-6C at 240. At least one of the palette windows in the hierarchy includes palette items (e.g., icons such as in Figures 5A-5E or textual indicators such as in Figures 6A-6C) that are selectable by a user to include functionality in a program being created or modified, e.g., program structures such as for and while loops, case statements, and global or local variables; operators such as numeric (addition, multiplication, etc.), comparison (equal to, greater than, etc.) and logical (AND, OR, XOR, etc) operators; file I/O items; signal or waveform processing function elements; complex mathematical function elements; instrument driver items; communications items; user interface elements for use in creating a user interface; nodes for use in creating a block diagram or graphical program; and/or graphical or sound items, among others. *See, e.g.,* Specification p. 19, lines 9-10; p. 20, lines 1-13; p. 23, line 21 – p. 24, line 26; Figure 5C at 212A; Figure 5E at 212B; Figure 6C at 248A.

The one or more palette windows include a palette window selection item that is selectable by the user to display a second palette window (e.g., a child palette window) from the hierarchy of palette windows. For instance, the palette window selection item may include a graphical or textual indication of the palette window (e.g., child palette window) that it represents. *See, e.g.,* Specification p. 19, lines 9-10; p. 20, lines 23-28; p. 23, line 21 – p. 25, line 2; p. 24, line 18 – p. 25, line 2; p. 28, lines 8-20; Figures 5A-5B at 208A; Figures 5C-5D at 208B; Figure 5E at 208C; Figure 6A at 244A.

The first palette window also includes one or more navigation items displayed on the first palette window for navigating among the hierarchy of palette windows. For example, the navigation items may include one or more of a back navigation item to open or display a most recent previously displayed palette window in a backward direction, (*See, e.g.,* Figures 5A, 5C, 5E, 6A, 6C at 216), a forward navigation item to open or display a most recently previously displayed palette window in a forward direction, (*See, e.g.,* Figures 5A, 5C, 5E, 6A, 6C at 218), and/or an up navigation item to open or display a parent palette window of the current palette window, (*See, e.g.,* Figures 5A, 5C, 5E, 6A, 6C at 220). *See, e.g.,* Specification p. 25, lines 9-16; p. 29, lines 1-8; Figures 5A-5E; Figures 6A-6C. Various other navigation items may be included which perform other types of navigation. *See, e.g.,* Specification p. 22, lines 3-9; p. 25, lines 18-26; p. 29, lines 10-17; Figures 5A-5E; Figures 6A-6C.

User input is received selecting a navigation item displayed on the first palette window. In response to the user input selecting the navigation item, the first palette window is closed or no longer displayed, and one of a parent palette window or a child palette window in relation to the first palette window in the hierarchy of palette windows is displayed. In one example, at most one palette window in a hierarchy is displayed at one time. For instance, when a previously displayed palette window, or one higher in the hierarchy, is opened using one of the navigation items, the current palette window is closed or no longer displayed. *See, e.g.,* Specification p. 22, lines 17-21; p. 25, line 20 – p. 27, line 10; p. 29, line 9 – p. 30 line 20.

Independent claim 95 is directed to a computer-implemented method for navigating among a hierarchy of palette windows in a graphical user interface displayed on a computer system, similar to that described above with respect to claim 73, but omitting the limitation "one or more of the palette windows comprise a palette window selection item, wherein the palette window selection item is selectable by the user do display a second palette window from the hierarch of palette windows". A currently displayed palette window from the hierarchy of palette windows is displayed on a display of the computer system, where one or more of the palette windows in the hierarchy include palette items that are selectable by a user to include functionality in a program being modified or created, as described above with reference to claim 73. *See, e.g.,* Specification p. 19, lines 9-10; p. 20, lines 1-13; p. 23, line 21 – p. 24, line 26; Figure 5C at 212A; Figure 5E at 212B; Figure 6C at 248A. The currently displayed palette window also includes one or more navigation items displayed on the currently displayed palette window for navigating among the hierarchy of palette windows. *See, e.g.,* p. 21, line 22 – p. 22, line 16, Figures 5A and 5C (elements 216, 218, 220). User input is received selecting a navigation item of the one or more navigation items displayed on the currently displayed palette window, in response to which a currently displayed palette window is closed, and a previously displayed palette window in response to said user input selecting the navigation item. *See, e.g.,* p. 22, lines 17-20.

Independent claim 117 is directed to a system suitable for implementing embodiments of a method similar to that of claim 73, described above, but omitting the limitation "one or more of the palette windows comprise a palette window selection item, wherein the palette window selection item is selectable by the user do display a second palette window from the hierarch of palette windows". The system includes a memory configured to store program instructions, an input device configured to receive user input, a display device (e.g., a monitor), and a processor (*See, e.g.,* p. 12, lines 14-17) configured to read the program instructions from the memory and to execute the program instructions to display on the display device a first palette window from a hierarchy of palette windows in a graphical user interface, where one or more of the palette windows in the hierarchy include palette items that are selectable by a user to include functionality in a program being created or modified. *See, e.g.,* Specification p. 14, line 26 – p. 15, line 1; p. 19, lines 9-10; p. 20, lines 1-13; p. 23, line 21 – p. 24, line 26; Figure 5C at 212A; Figure 5E at 212B; Figure 6C at 248A. The first palette window includes one or more navigation items displayed on the first palette window for navigating among the hierarchy of palette windows. (*See, e.g.,* p. 14, lines 1-10) User input is received from the input device selecting a navigation item displayed on the first palette window, in response to which the first palette window is closed, and at least one of a parent palette window or a child palette window in relation to the first palette window in the hierarchy of palette windows is displayed. *See, e.g.,* Specification p. 22, lines 17-21; p. 25, line 20 – p. 27, line 10; p. 29, line 9 – p. 30 line 20.

Independent claim 128 is directed to a system suitable for implementing embodiments of the method of claim 95, described above, where the system includes a memory configured to store program instructions, an input device configured to receive user input, a display device (e.g., a monitor), and a processor (*See, e.g.,* p. 12, lines 14-17) configured to read the program instructions from the memory and to execute the program instructions to perform various embodiments of the method. *See, e.g.,* Specification p. 19, lines 9-10; p. 20, lines 1-13; p. 23, line 21 – p. 24, line 26; Figure 5C at 212A; Figure 5E at 212B; Figure 6C at 248A; p. 21, line 22 – p. 22, line 16, Figures 5A and 5C (elements 216, 218, 220), and p. 22, lines 17-20.

Independent claim 143 is directed to a memory medium that stores program instructions executable to perform embodiments of the method of claim 95, described above. *See, e.g.,* Specification p. 19, lines 9-10; p. 20, lines 1-13; p. 23, line 21 – p. 24, line 26; Figure 5C at 212A; Figure 5E at 212B; Figure 6C at 248A; p. 21, line 22 – p. 22, line 16, Figures 5A and 5C (elements 216, 218, 220), and p. 22, lines 17-20.

Independent claim 149 is directed to a memory medium that stores program instructions executable to perform embodiments of the method of claim 95, described above. *See, e.g.,* Specification p. 19, lines 9-10; p. 20, lines 1-13; p. 23, line 21 – p. 24, line 26; Figure 5C at 212A; Figure 5E at 212B; Figure 6C at 248A; p. 21, line 22 – p. 22, line 16, Figures 5A and 5C (elements 216, 218, 220), and p. 22, lines 17-20.

Independent claim 104 is directed to a computer-implemented method for navigating among a hierarchy of palette windows in a graphical user interface displayed on a computer system, in which a first palette window from the hierarchy of palette windows is displayed on a display. *See, e.g.,* Specification p. 23, lines 15-20; p. 28, lines 3-7; Figures 5A-5E at 200; Figures 6A-6C at 240. One or more of the palette windows in the hierarchy include palette items (e.g., icons or textual indicators) that are selectable by a user to include functionality in a program being created or modified. *See, e.g.,* Specification p. 19, lines 9-10; p. 20, lines 1-13; p. 23, line 21 – p. 24, line 26; Figure 5C at 212A; Figure 5E at 212B; Figure 6C at 248A. The one or more palette windows also include one or more palette window selection items that are selectable by the user to display one or more different child palette windows (in the hierarchy of palette windows) of the first palette window. For instance, a palette window selection item of the one or more palette window selection items may include a graphical or textual indication of the palette child palette window that it represents. *See, e.g.,* Specification p. 19, lines 9-10; p. 20, lines 23-28; p. 23, line 21 – p. 25, line 2; p. 24, line 18 – p. 25, line 2; p. 28, lines 8-20; Figures 5A-5B at 208A; Figures 5C-5D at 208B; Figure 5E at 208C; Figure 6A at 244A. User input is received selecting a first palette window selection item of the one or more palette selection items in the first palette window. *See, e.g.,* Specification p. 20, lines 23-24; p. 24, lines 6-14; p. 28, lines 13-16. In response to the user input selecting the first palette window selection item, a second palette window which is a child palette window in relation to the first palette window in the hierarchy of palette windows is displayed, and the first palette window is closed or no longer displayed. In one example, at most one palette window in a hierarchy is displayed at one time. *See, e.g.*, Specification p. 20, line 25 – p. 21, line 6. The child palette window includes at least one palette item. *See, e.g.*, Specification p. 20, line 23 – p. 21, line 6.

Independent claim 136 is directed to a system suitable for implementing embodiments of the method described above with respect to claim 104. The system includes a memory configured to store program instructions, an input device configured to receive user input, a display device (e.g., a monitor), and a processor (*See, e.g.*, p. 12, lines 14-17) configured to read the program instructions from the memory and to execute the program instructions to display on the display device a first palette window from a hierarchy of palette windows in a graphical user interface, where one or more of the palette windows in the hierarchy include palette items that are selectable by a user to include functionality in a program being created or modified. *See, e.g.*, Specification p. 14, line 26 – p. 15, line 1; p. 19, lines 9-10; p. 20, lines 1-13; p. 23, line 21 – p. 24, line 26; Figure 5C at 212A; Figure 5E at 212B; Figure 6C at 248A. The first palette window includes one or more palette window selection items, each operable when selected to display a different child palette window of the first palette window in the hierarchy of palette windows. *See, e.g.*, p. 20, lines 23-28. User input is received from the input device selecting a first palette window selection item of the one or more selection items in the first palette window, in response to which a second palette window is displayed on the display device in response to the user input selecting the first palette window selection item. The second palette window is a child palette window of the first palette window in the hierarchy of palette windows. The first palette window is closed in response to the first user input selecting the first palette window selection item. *See, e.g.*, Specification p. 20, line 23 - p. 21, line 6.

Independent claim 151 is directed to a carrier medium that stores program instructions that are computer-executable to implement embodiments of the method described above with respect to claim 104, but omitting the limitation that "the child palette window comprises at least one palette item". In other words, in embodiments according to claim 151, it is not necessary that the child palette window include any palette items, e.g., for adding program functionality, i.e., the child window may be empty, or may only contain non-"palette item" elements. *See, e.g.,* Specification p. 20, line 23 - p. 21, line 15.

Independent claim 153 is directed to a computer-implemented method for creating or modifying a program using a hierarchy of palette windows in a graphical user interface displayed on a computer system. A first palette window from the hierarchy of palette windows is displayed on a display of the computer system, where the first palette window includes at least one palette window selection item that is selectable by a user. The palette window selection item is associated with a second palette window from the hierarchy of palette windows, where the second window from the hierarchy of palette windows includes at least one palette item that is selectable by the user, where the palette item is associated with functionality that can be included in the program being created or modified. In other words, the first palette window includes a user selectable item that invokes display of a second palette window that includes at least one user selectable palette item for including functionality in a program being created or modified. User input selecting the at least one palette window selection item from the first palette window is received, and in response, the second palette window is displayed and the first palette window is closed. *See, e.g.,* Specification p. 20, line 23 - p. 21, line 15.

VI. <u>GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL</u>

1. Claims 73-76, 78-80, 82-84, 86, 87, 95-101, 103-107, 109, 111-113, 117-120, 122, 124, 128-140, 143, 144, 146, 149, 151, 153-158 are rejected under 35 U.S.C.

103(a) as being unpatentable over Applicant's Admitted Prior Art (hereinafter, "AAPA") and Filepp et al. (U.S. Patent No. 5,758,072, hereinafter "Filepp").

2. Claims 81, 85, 88-94, 102, 108, 110, 114-116, 123, 125-127, 141, 142, 147, 148, 150, and 152 are rejected under 35 U.S.C. 103(a) as being unpatentable over AAPA, Filepp, and Gavron et al. (*How to Use Microsoft Windows NT 4 Workstation*, hereinafter "Gavron").

VII. ARGUMENT

First Ground of Rejection:

Claims 73-76, 78-80, 82-84, 86, 87, 95-101, 103-107, 109, 111-113, 117-120, 122, 124, 128-140, 143, 144, 146, 149, 151, 153-158 are finally rejected under 35 U.S.C. 103(a) as being unpatentable over AAPA and Filepp.

Appellant respectfully traverses this rejection for the following reasons. Different groups of claims are addressed under their respective subheadings.

Claims 73-76, 78, 80, 86, 87, 95-101, 103, 117-120, 124, 128-133, 135, 143, 144, 146, and 149

Appellant respectfully submits that each of the independent claims 73, 95, 117, 128, 143, and 149 recites one or more features not taught or suggested in AAPA and Filepp.

The Examiner admits that AAPA does not teach that a first palette window is closed subsequent to receiving user input selecting a navigation item (such as a "back", "up", or "next"), but asserts that combining AAPA with Filepp remedies this deficiency, citing Figure 3b of Filepp. Appellant respectfully disagrees.

Appellant respectfully notes "To establish *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. *In re*

Royka, 490 F.2d 981, 180 U.S.P.Q. 580 (C.C.P.A. 1974)" MPEP §2143.03 (*emphasis added*).

Appellant respectfully submits that Filepp is directed to:

> "a distributed processing, interactive computer network intended to provide very large numbers of simultaneous users; e.g. millions, with access to an interactive service having large numbers; e.g., thousands, of applications which include pre-created interactive text/graphic sessions; and more particularly, to a computer network in which the interactive text/graphic sessions are comprised of pre-created blocks of data and program instructions which may be distributed downwardly in the network for execution at software-enhanced user terminals..." (col. 1, lines 16-26)

Filepp's system includes "reception system" computers that "permit respective users to enter requests for interactive applications" (col. 3, lines 10-12), where the "Applications, i.e. information events, are composed of a sequence of one or more pages opened at screen 414 of monitor 412". The applications are "partitioned", in that they may be assembled (via pre-created objects) and provided to users dynamically. The cited portion of Filepp (Figure 3b) illustrates an example page for such an application, where the page has various partitions (distinct from "application partitions"), e.g., header (250), body (260), which may include display fields, an advertisement partition (280), and a command bar partition 190. Note that the page operates as a portion of an interactive application, specifically, a user-partitioned application interface (col. 69, lines 39-67) for the application. Each page may be considered a "screen" of the partitioned application, and may be built and provided to the user dynamically, based on user input.

As described in col. 70 lines 46-62, the various items included in the command bar 190 facilitate movement back and forth between, and operation of, the application pages. For example, "NEXT command 291 causes the next page in the current page set to be built." Similarly, the "BACK command 292 causes the previous page of the current page set to be built."

In other words, these navigation buttons allow the user to switch which application page, i.e., which portion of the application, is displayed on the monitor, e.g., to switch to a previous or next application page of the partitioned application. Nowhere does Filepp teach or suggest palette windows that include user selectable palette items for inclusion of desired functionality in a program, nor navigating among such palettes via navigation items included in the palette windows. Filepp specifically fails to teach or suggest "receiving user input selecting a navigation item displayed on the first palette window; closing the first palette window in response to said receiving user input selecting the navigation item; and displaying at least one of a parent palette window or a child palette window in relation to the first palette window in the hierarchy of palette windows in response to said user input selecting the navigation item."

Appellant submits that Filepp actually teaches *away* from Appellant's invention as claimed. Appellant respectfully reminds the Examiner that if a proposed modification would render the prior art feature unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification. In re Gordon, 733 F.2d 900 (Fed. Cir. 1984). Appellant submits that modifying Filepp to remedy the admitted deficiencies of AAPA would render Filepp's application pages unsatisfactory for their intended purpose. Appellant submits that, in fact, changing Filepp's application pages to palette windows would render Filepp's applications inoperable, since palettes cannot function as Filepp's executable applications. Applicant further submits that the Examiner has improperly selected only portions of Filepp's system using Appellant's claim as a guide or blueprint, ignoring the context and use of the selected portion with respect to the overall invention of Filepp.

Moreover, as stated in the MPEP §2143.01, "The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990). In addition, the showing of a suggestion, teaching, or motivation to combine prior teachings "must be clear and particular . . .. Broad conclusory statements regarding the teaching of multiple references, standing alone, are not 'evidence'." *In re Dembiczak*, 175 F.3d 994, 50 USPQ2d 1614 (Fed. Cir. 1999). The art must fairly teach or suggest to one to make the specific combination as claimed. That one achieves an improved result by making such a combination is no more than hindsight without an initial suggestion to make the combination.

Appellant submits that neither AAPA nor Filepp provides a motivation to combine. For example, Filepp nowhere mentions or even hints at the desirability of palette windows, nor navigating between such windows via Filepp's command bar buttons. Neither does AAPA suggest or indicate the desirability of this feature. Appellant notes that the only motivation to combine suggested by the Examiner is "being to easily navigate through the series of windows", which Appellant respectfully submits is simply citing an improved result without any initial suggestion from the prior art, which is improper. Thus, for at least the reasons provided above, Appellant submits that the Examiner's attempted combination of AAPA and Filepp is improper.

Appellant further submits that even were AAPA and Filepp properly combinable, which Appellant argues they are not, the resulting combination would still not produce Appellant's invention as claimed, as explained in detail above. More specifically, Appellant submits that the cited art does not teach or suggest: "receiving user input selecting a navigation item displayed on the first palette window; closing the first palette window in response to said receiving user input selecting the navigation item; and displaying at least one of a parent palette window or a child palette window in relation to the first palette window in the hierarchy of palette windows in response to said user input selecting the navigation item."

Thus, for at least the reasons provided above, Appellant submits that claim 73 and those claims dependent therefrom are patentably distinct and non-obvious over the AAPA and Filepp, taken singly or in combination, and are thus allowable. Claims 95, 117, 128, 143, and 149 include similar limitations as claim 73, and so the above arguments apply with equal force to these claims. Thus, Appellant submits that claims 95, 117, 128, 143, and 149, and those claims respectively dependent therefrom, are similarly patentably distinct and non-obvious over AAPA and Filepp, taken singly or in combination, and are thus allowable.

Claim 79

Appellant respectfully submits that neither AAPA nor Filepp, singly or in combination, teaches or suggests the features of claim 79. Claim 79, in addition to the features and limitations of claim 73, includes:

> ...wherein the one or more navigation items comprise one or more of a forward navigation item, a back navigation item, and an up navigation item.

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claim 79. For example, in addition to the arguments provided above with respect to independent claim 73, Appellant notes that since Filepp's partitioned application consists of a linear series of application pages, and is thus *not* organized in a logical tree structure, Filepp's system has no need for an "up" navigation item, which, as is well known, operates to move one step up in a tree structure. In fact, in Filepp's system, an "up" navigation item would be nonsensical and inoperable. Rather, Filepp's linear sequence of application pages uses "back" and "next" navigation features to move back and forth along the linear sequence of application pages, and makes no mention of an "up" navigation item. Thus, combining AAPA and Filepp (which Appellant has argued above is improper) would still not produce all the features and limitations of claim 79.

Thus, Appellant respectfully submits that AAPA and Filepp, taken singly or in combination, neither teach nor suggest all the features and limitations of claim 79, and so for at least the reasons provided above, claim 79 is patentably distinct and non-obvious over the cited art, and is thus allowable.

Claims 82, 83, and 84

Appellant respectfully submits that neither AAPA nor Filepp teaches or suggests the features of claims 82, 83, and 84. For example, claim 82, in addition to the features and limitations of claim 73, includes the limitation:

> ...wherein the one or more navigation items comprise one or more of a forward navigation menu item, a back navigation menu item, and an up navigation menu item.

In addition to the features and limitations of claim 73, claim 83 includes:

> ... wherein the one or more navigation items comprise a forward navigation menu item;
>
> wherein the forward navigation menu item is operable when selected to display a menu including one or more menu items

> each corresponding to a different previously displayed palette window in a forward direction;
>
> wherein each of the one or more menu items is operable when selected to display a previously displayed palette window corresponding to the selected menu item.

In addition to the features and limitations of claim 73, claim 84 includes:

> … wherein the one or more navigation items comprise a back navigation menu item;
>
> wherein the back navigation menu item is operable when selected to display a menu including one or more menu items each corresponding to a different previously displayed palette window in a backward direction;
>
> wherein each of the one or more menu items is operable when selected to display a previously displayed palette window corresponding to the selected menu item.

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claims 82, 83 and 84. For example, in addition to the arguments provided above with respect to independent claim 73, Appellant notes that Filepp fails to teach or suggest using menus to navigate among the pages of Filepp's application. For example, as Filepp describes in col. 70, lines 66-67, the "MENU command 294 causes the page presenting the previous set of choices to be rebuilt." In other words, Filepp's MENU command on the cited command bar 190 operates to rebuild an application page that presents a previous set of choices (e.g., provided by the user), *not* as a navigation tool to move back and forth between existing application pages, such functionality being provided by Filepp's BACK and NEXT buttons.

Thus, Appellant respectfully submits that AAPA and Filepp, taken singly or in combination, neither teach nor suggest all the features and limitations of claims 82, 83, and 84, and so for at least the reasons provided above, claims 82, 83, and 84 are patentably distinct and non-obvious over the cited art, and are thus allowable.

<u>Claims 104, 105, 106, 111, 112, 122, 136, 137, 138, 151, 153, 154, 155, 156, and 157</u>

Appellant respectfully submits that neither AAPA nor Filepp teaches or suggests the features of claims 104, 105, 106, 107, 109, 111, 112, 122, 136, 137, 138, 151, 153, 154, 155, 156, and 157. For example, claim 104 includes the limitations:

> ...receiving user input selecting a first palette window selection item of the one or more selection items in the first palette window;
>
> displaying a second palette window on the display in response to said receiving the user input selecting the first palette window selection item, wherein the second palette window is a child palette window of the first palette window in the hierarchy of palette windows, wherein the child palette window comprises at least one palette item; and
>
> closing the first palette window in response to said receiving the first user input selecting the first palette window selection item.

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claims 104, 105, 106, 107, 109, 111, 112, 122, 136, 137, 138, 151, 153, 154, 155, 156, and 157. For example, the Examiner asserts that AAPA teaches navigation items, citing element 108A of prior art Figures 4A, 4B, and 4C, as described on p. 1, lines 28-32, and p. 2, line 14 – p. 4, line 2. However, Appellant respectfully notes that the cited element 108A is not a navigation item, but rather is a palette window selection item, as clearly described the referenced portion of the specification. Appellant respectfully submits that throughout the present application, Appellant distinguishes, both via nomenclature and functional description, between "navigation items", e.g., for "back", "next", "up" movement relative to a current palette, and "palette window selection items", which serve to open respective ***specific*** child palettes with respect to a current parent palette. In other words, while a navigation item may switch from the current palette to any palette with the corresponding relationship indicated by the navigation item (e.g., "back"), each palette window selection item is specifically linked to or associated with a particular child palette of the current palette. Thus, Appellant submits that the Examiner's attempt to equate the two is improper.

Additionally, while AAPA does describe selection of a palette window selection item from a first palette window, and the subsequent opening of a respective child palette, as the Examiner admits, AAPA "fails to teach the closing of the first palette window and the displaying of the child palette window (second palette window) in response to a user input selection." The Examiner asserts that Filepp provides this missing functionality, citing col. 49, lines 39-41. Appellant respectfully disagrees.

The cited portion of Filepp recites:

> CLOSE_WINDOW [window-id];
> where, "window-id" contains the object ID of a new window to be opened after closing the currently open window.

Appellant submits that the citation actually describes a "verb" programming element in a text-based procedural language. For example, Filepp recites:

> Program objects 508 contain program instructions written in a high-level language called TRINTEX Basic Object Language, i.e., TBOL, described in greater detail hereafter, which may be executed on RS 400 to support the application. (col. 11, lines 33-37)

Appellant submits that the CLOSE_WINDOW [window-id] function cited by the Examiner is simply a description of a TBOL function callable in a text-based language to open a new window upon closing a currently open window, and that this text-based procedural language call in no way teaches the features and limitations of claim 104, specifically, "closing the first palette window in response to said receiving the first user input selecting the first palette window selection item."

Appellant further submits that, as stated in the MPEP §2143.01, "The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990). In addition, the showing of a suggestion, teaching, or motivation to combine prior teachings "must be clear and particular . . .. Broad conclusory statements regarding the teaching of multiple references, standing alone, are not 'evidence'." *In re Dembiczak*, 175 F.3d 994, 50 USPQ2d 1614 (Fed. Cir. 1999). The art must fairly teach or suggest to one to make the specific combination as claimed. That one achieves an improved result by making such a combination is no more than hindsight without an initial suggestion to make the combination.

Appellant submits that neither AAPA nor Filepp provides a motivation to combine. For example, Filepp nowhere mentions or even hints at the desirability of palette windows, nor the use and operation of such palette windows, e.g., "closing the first palette window in response to said receiving the first user input selecting the first palette window selection item". Neither does AAPA suggest or indicate the desirability of this feature. Appellant notes that the only motivation to combine suggested by the Examiner is "to enhance program execution efficiency", which Appellant respectfully submits is simply citing an improved result without any initial suggestion from the prior art, which is improper.

Moreover, Appellant submits that the Examiner's suggested motivation, *enhance efficiency*, is too general because it could cover almost any alteration contemplated of AAPA and does not address why this specific proposed modification would have been obvious. Additionally, there is nothing in either of references that would suggest "closing the first palette window in response to said receiving the first user input selecting the first palette window selection item". Finally, although Filepp discloses a single text-based procedural program call to close a current window and open a new window, there is no suggestion, other than Appellant's disclosure, to employ this scheme to promote the introduction of new and alternative products, specifically, to close a current palette window upon opening a new, e.g., child, window via selection of a palette window selection item in the current palette window. Thus, for at least the reasons provided above, Appellant submits that the Examiner's attempted combination of AAPA and Filepp is improper.

Appellant further submits that even were AAPA and Filepp properly combinable, which Appellant argues they are not, the resulting combination would still not produce Appellant's invention as represented in claim 104, as explained in detail above.

Thus, for at least the reasons provided above, Appellant respectfully submits that AAPA and Filepp, taken singly or in combination, neither teach nor suggest all the features and limitations of claim 104, and so for at least the reasons provided above, claim 104 and those claims dependent therefrom are patentably distinct and non-obvious over the cited art, and are thus allowable.

Independent claims 136, 151, and 153 include similar limitations as claim 104, being directed respectively to a system suitable for implementing embodiments of the method of 104, to a carrier medium storing program instructions implementing embodiments of the method of claim 104, and to a computer-implemented method with similar novel limitations, and so the above arguments apply with equal force to this claim. Thus, for at least the reasons provided above, Appellant submits that claims 136, 151, and 153, and those claims respectively dependent therefrom, are similarly patentably distinct and non-obvious over the cited art, and are thus allowable.

Claims 107, 109, 139, 140, and 158

Appellant respectfully submits that neither AAPA nor Filepp teaches or suggests the features of claims 107, 109, 139, and 140. For example, claim 107, in addition to the features and limitations of claim 104, includes the limitations:

> ... wherein each of the palette windows in the hierarchy of palette windows comprises one or more navigation items, the method further comprising:
>
> receiving user input selecting a navigation item of the second palette window, wherein the navigation item is operable when selected to close a currently displayed palette window and display a previously displayed palette window;
>
> closing the second palette window in response to said user input selecting the navigation item; and
>
> displaying the first palette window on the display in response to said user input selecting the navigation item.

Claim 158 includes similar limitations as claim 107.

Claim 139 includes similar limitations as claim 107, but specifies the navigation item as a "back navigation item operable when selected to display a most recently previously displayed palette window in a backward direction and to close a currently displayed palette window".

Claim 109 includes similar limitations as claim 107, but rather than displaying the first palette window on the display in response to said user input selecting the navigation item, claim 109 includes the limitation, "displaying a third palette window on the display in response to said user input selecting the navigation item".

Claim 140 includes similar limitations as claim 109, but specifies the navigation item as a "forward navigation item operable when selected to display a most recently previously displayed palette window in a forward direction and to close a currently displayed palette window".

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claims 107, 109, 139, 140, and 158. For example, in addition to the arguments provided above with respect to independent claim 104, Appellant notes that Filepp fails to teach or suggest "receiving user input selecting a navigation item of the second palette window, wherein the navigation item is operable when selected to close a currently displayed palette window and display a previously displayed palette window", and further fails to teach or suggest "closing the second palette window in response to said user input selecting the navigation item". Nor does Filepp teach or suggest the additional limitations of "displaying the first palette window on the display in response to said user input selecting the navigation item" (claim 107), nor "displaying a third palette window on the display in response to said user input selecting the navigation item" (claim 109)". Nor does Filepp teach or suggest these limitations where the navigation items are respectively specified as back and forward navigation items in the palette windows.

In rejecting claim 107, the Examiner asserts that AAPA teaches "when window 100B is closed, window 100A would be opened as part of the hierarchy window system". However, the specification clearly describes otherwise. For example, on p. 2, line 27 – p. 3, line 3, the present specification recites:

> Upon selecting palette window selection item 108A of palette window 100A, for example, by clicking or double-clicking the item, a second palette window 100B is displayed as illustrated in Figure 4B. In this example, palette window 100A may be described as a parent of palette window 100B, and palette window 100B as a child of palette window 100A. Palette window 100B may partially cover or overlap palette window 100A, which remains displayed.

Appellant respectfully notes that the Examiner's description of opening window 100A when closing window 100B is nowhere described in the AAPA. Rather, palette window 100B is displayed as a child palette window of palette window 100A in response to a user selecting palette window selection item 108A of palette window 100A. As indicated in the quoted passage above, in this prior art description, both palette windows 100A and 100B are displayed, albeit possibly in an overlapping manner.

Thus, for at least the reasons provided above, Appellant submits that AAPA and Filepp, taken singly or in combination, fail to teach or suggest all the limitations of claims 107, 109, 139, 140, and 158, and so submits that claims 107, 109, 139, 140, and 158 are patentably distinct and non-obvious over the cited art, and are thus allowable.

Claim 113

Appellant respectfully submits that neither AAPA nor Filepp teaches or suggests the features of claim 113. For example, claim 113, in addition to the features and limitations of claim 104, includes the limitation:

> ... wherein the second palette window comprises an up navigation item, wherein the up navigation item is operable when selected to display a parent palette window of the second palette window in the hierarchy of palette windows and to close the second palette window, the method further comprising:
>
> receiving user input selecting the up navigation item of the second palette window;
>
> displaying the first palette window on the display in response to said user input selecting the up navigation item; and
>
> closing the second palette window in response to said user input selecting the up navigation item.

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claim 113. For example, in addition to the arguments provided above with respect to independent claim 104, Appellant notes that Filepp fails to teach or suggest an "up" navigation item for moving upward in a hierarchy, i.e., in a tree structure, and specifically to disclose such a navigation item for navigating among a hierarchy of palette windows and described and claimed in the present application. In fact, as noted above, Filepp's navigation items (of command bar 190), "BACK" and "NEXT", are used to move back and forth between application pages of Filepp's page based application, where the pages compose a linear sequence, and as such do not form a hierarchical structure. In other words, there is no "up" direction with respect to the pages of Philepp's application. Thus, an "up" navigation item is neither needed nor accommodated in

Philepp's system. Appellant notes that Filepp's command bar 190 contains no "UP" navigation item, and in fact, such a navigation item is nowhere mentioned or discussed in Filepp.

Thus, for at least the reasons provided above, Appellant submits that AAPA and Filepp, taken singly or in combination, fail to teach or suggest all the limitations of claim 113, and so claim 113 is patentably distinct and non-obvious over the cited art, and is thus allowable.

Claim 134

Appellant respectfully submits that neither AAPA nor Filepp teaches or suggests the features of claim 134. Claim 134, in addition to the features and limitations of claim 128, includes:

> ... wherein the navigation item is an up navigation item operable when selected to display a parent palette window of the first palette window, regardless of the most recently previously displayed palette window.

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claim 134. For example, in addition to the arguments provided above with respect to independent claim 128, Appellant notes that since Filepp's partitioned application consists of a linear series of application pages, and is thus *not* organized in a logical tree structure, Filepp's system for navigating between application pages has no need for an "up" navigation item, which, as is well known, operates to move one step up in a hierarchical or tree structure. In fact, in Filepp's system, an "up" navigation item would be nonsensical and inoperable. Rather, Filepp's linear sequence of application pages uses "back" and "next" navigation features to move back and forth along the linear sequence of application pages, and makes no mention of an "up" navigation item. Appellant further submits that the notion of a "parent" is not applicable to Filepp's application pages, and in fact is never mentioned in Filepp. Thus, combining AAPA and Filepp (which Appellant has argued above is improper) would still not produce all the features and limitations of claim 134.

Thus, Appellant respectfully submits that AAPA and Filepp, taken singly or in combination, neither teach nor suggest all the features and limitations of claim 134, and so for at least the reasons provided above, claims 134 is patentably distinct and non-obvious over the cited art, and is thus allowable.

Second Ground of Rejection:

Claims 81, 85, 88-94, 102, 108, 110, 114-116, 123, 125-127, 141, 142, 147, 148, 150, and 152 are rejected under 35 U.S.C. 103(a) as being unpatentable over AAPA, Filepp, and Gavron.

Appellant respectfully traverses this rejection for the following reasons. Different groups of claims are addressed under their respective subheadings.

Claim 81, 102, 123, and 147

Appellant respectfully submits that neither AAPA, Filepp, nor Gavron teaches or suggests the features of claim 81. Claim 81, in addition to the features and limitations of claim 73, includes:

> ...wherein the one or more navigation items comprise an up navigation item;
>
> wherein the up navigation item is operable when selected to display a parent palette window of the first palette window, regardless of a most recently previously displayed palette window.

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claims 81, 102, 123, and 147. For example, the Examiner admits that AAPA and Filepp "fail to teach an 'up' icon for the purpose of enabling user [sic] to easily navigate through a hierarchy of windows", but asserts that Gavron remedies the admitted deficiencies of AAPA and Filepp, citing page 41 of Gavron. Appellant respectfully disagrees.

Appellant notes that pages 40 and 41 of Gavron are directed to "How to Copy and Move Files and Folders" using Microsoft Corporation's Windows Explorer tool. More specifically, the "up" icon cited by the Examiner is clearly (and only) operable to move upward in a directory structure of a computer system. Nowhere does Gavron mention or even hint at palette windows as described and claimed in the present application.

Moreover, Appellant submits that Gavron fails to provide a motivation to combine with AAPA (and/or Filepp), since Gavron fails to address palette windows at all. Appellant notes that the only motivation to combine suggested by the Examiner is "to enable user [sic] to easily navigate through a hierarchy of windows", which Appellant respectfully submits is simply citing an improved result, absent any motivation provided by the prior art, which is hindsight analysis, and thus improper.

Thus, Appellant submits that the Examiner's attempted combination of AAPA (and/or Filepp) and Gavron is improper, and so cannot be used to establish a 103(a) rejection. Additionally, Appellant submits that even were AAPA (and/or Filepp) and Gavron properly combinable, which Appellant argues they are not, the alleged combination would still not produce Appellant's invention as represented in claim 81 (as well as 102, 123, and 147). For example, the Examiner has not indicated how to modify Gavron from navigating in a computer file system's directory structure to navigating (and displaying) a hierarchy of palettes, as described and claimed by Appellant. Appellant respectfully submits that simply combining Gavron's directory navigation system ("up" icon) with AAPA would not provide the claimed functionality of moving from a current palette to a parent palette in a hierarchy of palettes, and displaying the parent palette (and automatically closing the current palette). Rather, selecting Gavron's up icon would simply move from a current directory to the directory above, which is not germane to palette navigation. Thus, Appellant submits that Gavron actually teaches *away* from Appellant's invention as claimed.

Regarding the Filepp reference, in addition to the arguments provided above with respect to independent claim 73, 117, and 136, Appellant notes that since Filepp's partitioned application consists of a linear series of application pages, and is thus *not* organized in a logical tree structure, Filepp's system has no need for an "up" navigation item, which, as is well known, operates to move one step up in a tree structure. In fact, in Filepp's system, an "up" navigation item would be nonsensical and inoperable. Rather, Filepp's linear sequence of application pages uses "back" and "next" navigation features to move back and forth along the linear sequence of application pages, and makes no mention of an "up" navigation item. Appellant further notes, as argued above, that

Filepp's application pages are not palette windows as described and claimed in the present application.

Thus, combining AAPA, Filepp, and Gavron would still not produce all the features and limitations of claim 81.

Thus, Appellant respectfully submits that AAPA, Filepp, and Gavron, taken singly or in combination, neither teach nor suggest all the features and limitations of claim 81, and so for at least the reasons provided above, claims 81 is patentably distinct and non-obvious over the cited art, and is thus allowable.

Claims 102, 123, and 147 include similar limitations as claim 81, and so the above arguments apply with equal force to these claims. Thus, for at least the reasons provided above, Appellant submits that claims 102, 123, and 147 are patentably distinct and non-obvious over the cited art, and are thus allowable.

Claim 85

Appellant respectfully submits that neither AAPA nor Filepp nor Gavron teaches or suggests the features of claim 85. For example, in addition to the features and limitations of claim 73, claim 85 includes:

> ... wherein the one or more navigation items comprise an up navigation menu item;
>
> wherein the up navigation menu item is operable when selected to display a menu including one or more menu items each corresponding to a different palette window above the first palette window in the hierarchy of palette windows;
>
> wherein each of the one or more menu items is operable when selected to display a palette window corresponding to the selected menu item.

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claim 85. The Examiner admits that AAPA and Filepp "fail to teach an 'up' icon for the purpose of enabling user [sic] to easily navigate through a hierarchy of windows", but asserts that Gavron remedies the admitted deficiencies of AAPA and Filepp, citing page 41 of Gavron. Appellant respectfully disagrees.

As noted above with respect to claims 81, 102, 123, and 147, pages 40 and 41 of Gavron are directed to "How to Copy and Move Files and Folders" using Microsoft

Corporation's Windows Explorer tool, where the "up" icon cited by the Examiner is for moving upward in a computer system's directory structure. Gavron nowhere mentions or even hints at palette windows as described and claimed in the present application. Appellant further notes that Gavron's "up" icon is specifically *not* a navigation *menu* item, and submits that the Examiner's attempt to equate the two is improper.

Appellant also submits that Gavron also fails to provide a motivation to combine with AAPA (and/or Filepp), since Gavron fails to describe or address palette windows at all, and makes no mention of using a navigation menu item to navigate upward in a palette window hierarchy. As noted above, the only motivation to combine suggested by the Examiner is "to enable user [sic] to easily navigate through a hierarchy of windows", which Appellant respectfully submits is simply citing an improved result, absent any motivation provided by the prior art, which is hindsight analysis, and thus improper. Thus, Appellant submits that the Examiner's attempted combination of AAPA (and/or Filepp) and Gavron is improper, and so cannot be used to establish a 103(a) rejection.

Additionally, Appellant submits that even were AAPA (and/or Filepp) and Gavron properly combinable, which Appellant argues they are not, the alleged combination would still not produce Appellant's invention as represented in claim 85. For example, the Examiner has not indicated how to modify Gavron's up icon for navigating in a computer file system's directory structure to an up navigation menu item for navigating (and displaying) a hierarchy of palettes, as described and claimed by Appellant. Appellant respectfully submits that simply combining Gavron's directory navigation system ("up" icon) with AAPA would not provide the claimed functionality of a selected up navigation *menu* item moving from a current palette to a parent palette in a hierarchy of palettes, and displaying the parent palette (and automatically closing the current palette). Rather, selecting Gavron's up icon would simply move from a current directory to the directory above, which is not germane to palette navigation. Thus, Appellant submits that Gavron actually teaches *away* from Appellant's invention as claimed. Finally, Appellant notes that even if one were to modify Gavron to navigate among palette windows, Gavron would then be rendered unsatisfactory for its intended purpose, which is to allow a user to navigate in a directory structure.

Thus, Appellant submits that Gavron fails to remedy the admitted deficiencies of AAPA (and/or Filepp).

In addition to the arguments provided above with respect to independent claim 73, Appellant notes that Filepp fails to teach or suggest using menus to navigate among the pages of Filepp's application. For example, as Filepp describes in col. 70, lines 66-67, the "MENU command 294 causes the page presenting the previous set of choices to be rebuilt." In other words, Filepp's MENU command on the cited command bar 190 operates to rebuild an application page that presents a previous set of choices (e.g., provided by the user), *not* as a navigation tool to move back and forth between existing application pages, such functionality being provided by Filepp's BACK and NEXT buttons.

Additionally, specifically with respect to claim 85, Appellant notes that since Filepp's page based application is a linear sequence of pages, i.e., is not a hierarchical tree structure, no "up" navigation menu item is disclosed by Filepp, and additionally, as noted above, such functionality is not needed, or even accommodated, in Filepp's system.

Thus, Appellant respectfully submits that AAPA, Filepp, and Gavon, taken singly or in combination, neither teach nor suggest all the features and limitations of claims 82, 83, 84, and 85, and so for at least the reasons provided above, claims 82, 83, 84, and 85 are patentably distinct and non-obvious over the cited art, and are thus allowable.

Claims 88, 89, 90, 91, 92, 93, 94, and 148

Appellant respectfully submits that neither AAPA nor Filepp nor Gavron teaches or suggests the features of claims 88, 89, 90, 91, 92, 93, 94, and 148. For example, in addition to the features and limitations of claim 73, claim 88 includes the limitations:

> … wherein each of the palette windows in the hierarchy of palette windows comprises a search item, the method further comprising:
>
> receiving user input selecting a search item of a currently displayed palette window;
>
> displaying a search window in response to said user input selecting the search item;
>
> receiving user input in the search window specifying a search criteria;
>
> identifying a new palette window in the search window in accordance with the search criteria user input; and

displaying the new palette window.

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claim 88.

The Examiner admits that AAPA (and apparently Filepp) fails to teach the search features and limitations of claim 88, but asserts that Gavron provides these missing features and limitations, citing page 7. Appellant respectfully disagrees.

Appellant notes that the cited portion of Gavron actually describes a search feature in a Help tool for Microsoft's NT operating system, where a user enters text describing a desired help topic, and the tool queries a help topic database and returns any positive results, i.e., text addressing the help topic. Nowhere does Gavron (or the Examiner) indicate how such a general text-based search tool may be modified to operate specifically in the realm of palette windows as described and claimed in the present application. Nor can Appellant find any mention or even hint in Gavron of providing a search capability for a palette window hierarchy. Appellant further notes that Gavron's search feature locates help text stored in or accessed via a help text database, and submits that this functionality is not immediately applicable to searching for and identifying a palette window in accordance with user-supplied search criteria.

Appellant further submits that modifying Gavron to search for palette windows as described in the present application would render Gavron's search tool incapable of performing its intended function—performing text-based searches for text-based help information. Appellant also submits that neither AAPA, Filepp, nor Gavron provides a motivation to combine, noting that the only suggestion to combine provided by the Examiner is "to provide convenient searching feature" [sic], which Appellant respectfully submits is simply citing an improved result, absent any motivation found in the prior art. Thus, for at least these reasons, Appellant submits that the Examiner's attempt to combine AAPA, Filepp, and Gavron is improper. Appellant further submits that even were AAPA, Filepp, and Gavron properly combinable, which Appellant argues they are not, the resulting combination would still not produce Appellant's invention as represented in claim 88, as argued above.

Thus, for at least the reasons provided above, Appellant submits that AAPA, Filepp, and Gavron, taken singly or in combination, fail to teach or suggest all the features and limitations of claim 88, and so claim 88 and those claims dependent therefrom are patentably distinct and non-obvious over AAPA, Filepp, and Gavron, and are thus allowable.

Claim 148 includes similar limitations as claim 88, and so the above arguments apply with equal force to this claim. Thus, for at least the reasons provided above, Appellant submits that claim 148 is patentably distinct and non-obvious over AAPA, Filepp, and Gavron, taken singly, or in combination, and is thus allowable.

Claims 108 and 110

Appellant respectfully submits that neither AAPA nor Filepp nor Gavron teaches or suggests the features of claims 108 and 110. For example, in addition to the features and limitations of claim 104, claim 108 includes the limitations:

> ... wherein the navigation item is a back navigation item operable when selected to display a most recently previously displayed palette window in a backward direction.

In addition to the features and limitations of claim 104, claim 110 includes the limitations:

> ... wherein the navigation item is a forward navigation item operable when selected to display a most recently previously displayed palette window in a forward direction.

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claims 108 and 110.

The Examiner's arguments appear to be directed to search functionality, which is not germane to the subject matter of claims 108 and 110. More specifically, the Examiner admits that AAPA and apparently Filepp fail to teach the search features and limitations of claim 108 and 110, but asserts that Gavron provides these missing features and limitations, citing page 7 of Gavron. Appellant respectfully disagrees, and submits that claims 108 and 110 should have been grouped and addressed in the first ground of rejection above, in which case Appellant would have added these two claims to the claim group that includes claims 104, 105, 106, 107, 109, 111, 112, 122, 136, 137, 138, 151, 153, 154, 155, 156, and 157.

However, Appellant respectfully submits that since the cited portions of Gavon, and the Examiner's arguments appear to address subject matter unrelated to claims 108 and 110, a *prima facie* case of obviousness has not been established to reject claims 108 and 110.

Thus, for at least the reasons provided above, Appellant submits that AAPA, Filepp, and Gavron, taken singly or in combination, fail to teach or suggest all the features and limitations of claim 108 and 110, and so claims 108 and 110 are patentably distinct and non-obvious over the cited art, and are thus allowable.

<u>Claims 114 – 116, 142, and 152</u>

Appellant respectfully submits that neither AAPA nor Filepp nor Gavron teaches or suggests the features of claims 114 – 116, 142, and 152. For example, in addition to the features and limitations of claim 104, claim 114 includes the limitations:

> ... wherein each of the palette windows in the hierarchy of palette windows comprises a search item, the method further comprising:
>
> receiving user input selecting a search item of a currently displayed palette window;
>
> closing the currently displayed palette window in response to said user input selecting the search item;
>
> displaying a search window in response to said user input selecting the search item;
>
> receiving user input in the search window specifying a search criteria;
>
> identifying a new palette window in the search window in accordance with the search criteria user input;
>
> closing the search window; and
>
> displaying the new palette window.

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claim 114.

The Examiner admits that AAPA (and apparently Filepp) fails to teach the search features and limitations of claim 114, but asserts that Gavron provides these missing features and limitations, citing page 7. Appellant respectfully disagrees.

As argued above with respect to claim 88, Appellant notes that the cited portion of Gavron actually describes a search feature in a Help tool for Microsoft's NT operating system, where a user enters text describing a desired help topic, and the tool queries a help topic database and returns any positive results, i.e., text addressing the help topic. Nowhere does Gavron (or the Examiner) indicate how such a text-based search tool may be modified to operate specifically in the realm of palette windows as described and claimed in the present application. Nor can Appellant find any mention or even hint in Gavron of providing a search capability for a palette window hierarchy. Appellant further notes that Gavron's search feature locates help text stored in or accessed via a help text database, and submits that this functionality is not immediately applicable to searching for and identifying a palette window in accordance with user-supplied search criteria (and displaying the identified palette).

Appellant also notes that claim 114 includes the limitations "closing the currently displayed palette window in response to said user input selecting the search item" and "closing the search window", which is also not taught or suggested by Gavron (or AAPA or Filepp).

Appellant further submits that modifying Gavron to search for palette windows as described in the present application would render Gavron's search tool incapable of performing its intended function—performing text-based searches for text-based help information. Additionally, Appellant notes that prior art text-based search tools such as Gavon describes do not generally close a current file, directory, or document upon opening a search window, nor close the search window upon finding matching text, and submits that were Gavron modified to include this feature, users would almost certainly be frustrated to find that their original (current) file, directory, or document window were automatically closed.

Appellant also submits that neither AAPA, Filepp, nor Gavron provides a motivation to combine, noting that the only suggestion to combine provided by the Examiner is "to provide convenient searching feature" [sic], which Appellant respectfully submits is simply citing an improved result, absent any motivation found in the prior art. Thus, for at least these reasons, Appellant submits that the Examiner's attempt to combine AAPA, Filepp, and Gavron is improper. Appellant further submits that even were AAPA, Filepp, and Gavron properly combinable, which Appellant argues they are not, the resulting combination would still not produce Appellant's invention as represented in claim 114, as argued above.

Claim 142 includes similar limitations as claim 114, and so the above arguments apply with equal force to this claim.

Thus, for at least the reasons provided above, Appellant submits that AAPA, Filepp, and Gavron, taken singly or in combination, fail to teach or suggest all the features and limitations of claims 114, 142, and 152, and so claims 114, 142, and 152, and those claims respectively dependent therefrom, are patentably distinct and non-obvious over AAPA, Filepp, and Gavron, and are thus allowable.

__Claims 125 – 127__

Appellant respectfully submits that neither AAPA nor Filepp nor Gavron teaches or suggests the features of claims 125 – 127. For example, in addition to the features and limitations of claim 117, claim 125 includes the limitations:

> ... wherein each of the palette windows in the hierarchy of palette windows comprises a search item, wherein, in response to execution of the program instructions, the processor is further operable to:
>
> receive user input selecting a search item of a currently displayed palette window;
>
> close the currently displayed palette window in response to said user input selecting the search item;
>
> display a search window on the display device in response to said user input selecting the search item;
>
> receive user input in the search window specifying a search criteria;
>
> identify a new palette window in the search window in accordance with the search criteria user input;
>
> close the search window; and
>
> display the new palette window on the display device.

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claim 125.

The Examiner admits that AAPA (and apparently Filepp) fails to teach the search features and limitations of claim 125, but asserts that Gavron provides these missing features and limitations, citing page 7. Appellant respectfully disagrees.

As argued above with respect to claim 125, Appellant notes that the cited portion of Gavron actually describes a search feature in a Help tool for Microsoft's NT operating system, where a user enters text describing a desired help topic, and the tool queries a help topic database and returns any positive results, i.e., text addressing the help topic. Nowhere does Gavron (or the Examiner) indicate how such a text-based search tool may be modified to operate specifically in the realm of palette windows as described and claimed in the present application. Nor can Appellant find any mention or even hint in Gavron of providing a search capability for a palette window hierarchy. Appellant further notes that Gavron's search feature locates help text stored in or accessed via a help text database, and submits that this functionality is not immediately applicable to searching for and identifying a palette window in accordance with user-supplied search criteria (and displaying the identified palette).

Appellant also notes that claim 125 includes the limitations "closing the currently displayed palette window in response to said user input selecting the search item" and "closing the search window", which is also not taught or suggested by Gavron (or AAPA or Filepp).

Appellant further submits that modifying Gavron to search for palette windows as described in the present application would render Gavron's search tool incapable of performing its intended function—performing text-based searches for text-based help information. Additionally, Appellant notes that prior art text-based search tools such as Gavon describes do not generally close a current file, directory, or document upon opening a search window, nor close the search window upon finding matching text, and submits that were Gavron modified to include this feature, users would almost certainly be frustrated to find that their original (current) file, directory, or document window were automatically closed.

Appellant also submits that neither AAPA, Filepp, nor Gavron provides a motivation to combine, noting that the only suggestion to combine provided by the Examiner is "to provide convenient searching feature" [sic], which Appellant respectfully submits is simply citing an improved result, absent any motivation found in the prior art. Thus, for at least these reasons, Appellant submits that the Examiner's attempt to combine AAPA, Filepp, and Gavron is improper. Appellant further submits that even were AAPA, Filepp, and Gavron properly combinable, which Appellant argues they are not, the resulting combination would still not produce Appellant's invention as represented in claim 125, as argued in detail above.

Thus, for at least the reasons provided above, Appellant submits that AAPA, Filepp, and Gavron, taken singly or in combination, fail to teach or suggest all the features and limitations of claim 125, and so claim 125 and those claims respectively dependent therefrom are patentably distinct and non-obvious over AAPA, Filepp, and Gavron, and are thus allowable.

Claims 141

Appellant respectfully submits that neither AAPA nor Filepp teaches or suggests the features of claim 141. For example, claim 141, in addition to the features and limitations of claim 136, includes the limitation:

> ... wherein the second palette window comprises an up navigation item, wherein the up navigation item is operable when selected to display a parent palette window of a currently displayed palette window in the hierarchy of palette windows and to close the currently displayed palette window, wherein, in response to execution of the program instructions, the processor is further operable to:
>
> receive user input from the input device selecting the up navigation item of the second palette window;
>
> display the first palette window on the display device in response to said user input selecting the up navigation item; and
>
> close the second palette window in response to said user input selecting the up navigation item.

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claim 141. For example, in addition to the arguments provided above with respect to independent claim 136, Appellant notes that Filepp fails to teach or suggest an "up" navigation item for moving upward in a hierarchy, i.e., in a tree structure, and specifically to disclose such a navigation item for navigating among a hierarchy of palette windows and described and claimed in the present application. In fact, as argued above with respect to claim 113, Filepp's navigation items (of command bar 190), "BACK" and "NEXT", are used to move back and forth between application pages of Filepp's page based application, where the pages compose a linear sequence, and as such do not form a hierarchical structure, i.e., there is no "up" direction with respect to the pages of Philepp's application. Thus, an "up" navigation item is neither needed nor accommodated in Philepp's system. Appellant notes that Filepp's command bar 190 contains no "UP" navigation item, and in fact, such a navigation item is nowhere mentioned or discussed in Filepp.

Thus, for at least the reasons provided above, Appellant submits that AAPA and Filepp, taken singly or in combination, fail to teach or suggest all the limitations of claim 141, and so claim 141 is patentably distinct and non-obvious over the cited art, and is thus allowable.

Claim 150

Appellant respectfully submits that neither AAPA nor Filepp nor Gavron teaches or suggests the features of claim 150. For example, in addition to the features and limitations of independent claim 149, claim 150 includes the limitation:

> ... wherein the navigation item is one of a back navigation item, a forward navigation item, and an up navigation item..

Appellant respectfully submits that a *prima facie* case of obviousness has not been established to reject claim 150. For example, as argued above with respect to claim the Examiner admits that AAPA and Filepp "fail to teach an 'up' icon for the purpose of enabling user [sic] to easily navigate through a hierarchy of windows", but asserts that Gavron remedies the admitted deficiencies of AAPA and Filepp, citing page 41 of Gavron. Appellant respectfully disagrees.

Appellant notes that pages 40 and 41 of Gavron are directed to "How to Copy and Move Files and Folders" using Microsoft Corporation's Windows Explorer tool. More specifically, the "up" icon cited by the Examiner is clearly (and only) operable to move upward in a directory structure of a computer system. Nowhere does Gavron mention or even hint at palette windows as described and claimed in the present application.

Moreover, Appellant submits that Gavron fails to provide a motivation to combine with AAPA (and/or Filepp), since Gavron fails to address palette windows at all. Appellant notes that the only motivation to combine suggested by the Examiner is "to enable user [sic] to easily navigate through a hierarchy of windows", which Appellant respectfully submits is simply citing an improved result, absent any motivation provided by the prior art, which is hindsight analysis, and thus improper.

Thus, Appellant submits that the Examiner's attempted combination of AAPA (and/or Filepp) and Gavron is improper, and so cannot be used to establish a 103(a) rejection. Additionally, Appellant submits that even were AAPA (and/or Filepp) and Gavron properly combinable, which Appellant argues they are not, the alleged combination would still not produce Appellant's invention as represented in claim 150. For example, the Examiner has not indicated how to modify Gavron from navigating in a computer file system's directory structure to navigating (and displaying) a hierarchy of palettes, as described and claimed by Appellant. Appellant respectfully submits that simply combining Gavron's directory navigation system ("up" icon) with AAPA would not provide the claimed functionality of closing the current palette window and displaying the previously displayed palette window. Rather, selecting Gavron's up icon would simply move from a current directory to the directory above, which is not germane to palette navigation. Thus, Appellant submits that Gavron actually teaches *away* from Appellant's invention as claimed. Thus, Appellant respectfully submits that Gavron fails to remedy the admitted deficiencies of AAPA and Filepp.

Appellant further notes that Filepp's back and forward navigation buttons operate to move back and forth among Filepp's application pages, and thus would not be operable to navigate between palette windows in a hierarchy of palette windows.

Thus, combining AAPA, Filepp, and Gavron would still not produce all the features and limitations of claim 150.

Thus, Appellant respectfully submits that AAPA, Filepp, and Gavron, taken singly or in combination, neither teach nor suggest all the features and limitations of claim 150, and so for at least the reasons provided above, claim 150 is patentably distinct and non-obvious over the cited art, and is thus allowable.

VIII. CONCLUSION

For the foregoing reasons, it is submitted that the Examiner's rejection of claims 1-40 was erroneous, and reversal of Examiner's decision is respectfully requested.

The Commissioner is authorized to charge the appeal brief fee of $500.00 and any other fees that may be due to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5150-43100/JCH. This Appeal Brief is submitted with a return receipt postcard.

Respectfully submitted,

Jeffrey C. Hood
Reg. No. 35,198
Attorney for Appellant(s)

Meyertons, Hood, Kivlin, Kowert & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
(512) 853-8800

Date: May 20, 2005 JCH/IMF/MSW

IX. CLAIMS APPENDIX

The claims on appeal are as follows.

Claims 1-72 (Canceled)

73. A computer-implemented method for navigating among a hierarchy of palette windows in a graphical user interface displayed on a computer system, wherein the computer system includes a display, the method comprising:

displaying on the display a first palette window from the hierarchy of palette windows, wherein one or more of the palette windows in the hierarchy comprise palette items that are selectable by a user, wherein each of the palette items is selectable by the user to include functionality in a program being created or modified, wherein one or more of the palette windows comprise a palette window selection item, wherein the palette window selection item is selectable by the user to display a second palette window from the hierarchy of palette windows, and wherein the first palette window includes one or more navigation items displayed on the first palette window for navigating among the hierarchy of palette windows;

receiving user input selecting a navigation item displayed on the first palette window;

closing the first palette window in response to said receiving user input selecting the navigation item; and

displaying at least one of a parent palette window or a child palette window in relation to the first palette window in the hierarchy of palette windows in response to said user input selecting the navigation item.

74. The method of claim 73, wherein at least one of the palette items includes an icon that is selectable by the user to include the functionality associated with the palette item in the program.

75. The method of claim 73, wherein at least one of the palette items includes an icon that is selectable by the user to incorporate a Graphical User Interface (GUI) element in a GUI of the program being created or modified.

76. The method of claim 73, wherein the program being created or modified is a graphical program, and wherein the palette items include icons that are each selectable by the user to include a node in the graphical program.

77. (Canceled)

78. The method of claim 73, wherein the navigation item is operable when selected to close a currently displayed palette window and display a previously displayed palette window.

79. The method of claim 73, wherein the one or more navigation items comprise one or more of a forward navigation item, a back navigation item, and an up navigation item.

80. The method of claim 73, wherein the one or more navigation items comprise a forward navigation item and a back navigation item;

wherein the forward navigation item is operable when selected to display a most recently previously displayed palette window in a forward direction;

wherein the back navigation item is operable when selected to display a most recently previously displayed palette window in a backward direction.

81. The method of claim 73, wherein the one or more navigation items comprise an up navigation item;

wherein the up navigation item is operable when selected to display a parent palette window of the first palette window, regardless of a most recently previously displayed palette window.

82. The method of claim 73, wherein the one or more navigation items comprise one or more of a forward navigation menu item, a back navigation menu item, and an up navigation menu item.

83. The method of claim 73, wherein the one or more navigation items comprise a forward navigation menu item;

wherein the forward navigation menu item is operable when selected to display a menu including one or more menu items each corresponding to a different previously displayed palette window in a forward direction;

wherein each of the one or more menu items is operable when selected to display a previously displayed palette window corresponding to the selected menu item.

84. The method of claim 73, wherein the one or more navigation items comprise a back navigation menu item;

wherein the back navigation menu item is operable when selected to display a menu including one or more menu items each corresponding to a different previously displayed palette window in a backward direction;

wherein each of the one or more menu items is operable when selected to display a previously displayed palette window corresponding to the selected menu item.

85. The method of claim 73, wherein the one or more navigation items comprise an up navigation menu item;

wherein the up navigation menu item is operable when selected to display a menu including one or more menu items each corresponding to a different palette window above the first palette window in the hierarchy of palette windows;

wherein each of the one or more menu items is operable when selected to display a palette window corresponding to the selected menu item.

86. The method of claim 73,

wherein the first palette window comprises one or more palette window selection items, wherein each of the one or more palette window selection items is operable when selected to display a different child palette window of the first palette window in the hierarchy of palette windows.

87. The method of claim 73, further comprising:

prior to said displaying the first palette window,

displaying on the display a first parent palette window from the hierarchy of palette windows, wherein the first parent palette window is a parent of the first palette window, wherein the first parent window comprises a palette window selection item which corresponds to the first palette window;

receiving user input selecting a palette window selection item which corresponds to the first palette window from the first parent palette window;

wherein said displaying the first palette window is performed in response to said user input selecting the palette window selection item which corresponds to the first palette window;

wherein said displaying at least one of a parent palette window or a child palette window comprises displaying the first parent palette window.

88. The method of claim 73, wherein each of the palette windows in the hierarchy of palette windows comprises a search item, the method further comprising:

receiving user input selecting a search item of a currently displayed palette window;

displaying a search window in response to said user input selecting the search item;

receiving user input in the search window specifying a search criteria;

identifying a new palette window in the search window in accordance with the search criteria user input; and

displaying the new palette window.

89. The method of claim 88, wherein said identifying comprises:

identifying and displaying information regarding a plurality of possible palette windows in the search window in accordance with the search criteria; and receiving user input selecting the new palette window from the plurality of possible palette windows.

90. The method of claim 89, wherein the user input in the search window specifying a search criteria includes a search string, and wherein said identifying and displaying information regarding a plurality of possible palette windows in the search window in accordance with the search criteria comprises:

searching for the search string in a plurality of text items related to palette windows in the hierarchy; and

displaying one or more located text items in the search window, wherein each of the one or more located text items includes the search string, and wherein each of the one or more located text items references one of the plurality of possible palette windows.

91. The method of claim 90, wherein the user input selecting the new palette window from the plurality of possible palette windows specifies one of the one or more located text items in the search window, wherein the specified located text item references the new palette window.

92. The method of claim 89, further comprising closing the search window subsequent to said receiving user input selecting the new palette window.

93. The method of claim 89, wherein the plurality of possible palette windows includes palette windows from a plurality of hierarchies of palette windows.

94. The method of claim 89, further comprising closing the currently displayed palette window subsequent to said receiving user input selecting the search item of the currently displayed palette window.

95. A computer-implemented method for navigating among a hierarchy of palette windows in a graphical user interface displayed on a computer system, wherein the computer system includes a display, the method comprising:

displaying on the display a currently displayed palette window from the hierarchy of palette windows, wherein one or more of the palette windows in the hierarchy comprise palette items that are selectable by a user to include functionality in a program being modified or created, and wherein the currently displayed palette window includes one or more navigation items displayed on the currently displayed palette window for navigating among the hierarchy of palette windows;

receiving user input selecting a navigation item of the one or more navigation items displayed on the currently displayed palette window;

closing a currently displayed palette window in response to said user input selecting the navigation item; and

displaying a previously displayed palette window in response to said user input selecting the navigation item.

96. The method of claim 95, wherein the palette items include icons that are selectable by the user to incorporate Graphical User Interface (GUI) elements in a GUI of the program.

97. The method of claim 95, wherein the program being created or modified is a graphical program, and wherein the palette items include icons that are selectable by the user to include nodes in the graphical program.

98. The method of claim 95, wherein the navigation item is operable when selected to close the currently displayed palette window and display a previously displayed palette window.

99. The method of claim 95, wherein the navigation item is one of a forward navigation item, a backward navigation item, and an up navigation item.

100. The method of claim 95, wherein the navigation item is a back navigation item operable when selected to display a most recently previously displayed palette window in a backward direction.

101. The method of claim 95, wherein the navigation item is a forward navigation item operable when selected to display a most recently previously displayed palette window in a forward direction.

102. The method of claim 95, wherein the navigation item is an up navigation item operable when selected to display a parent palette window of the first palette window, regardless of the most recently previously displayed palette window.

103. The method of claim 95,

wherein the first palette window comprises one or more palette window selection items, wherein each of the one or more palette window selection items is operable when selected to display a different child palette window of the first palette window in the hierarchy of palette windows.

104. A computer-implemented method for navigating among a hierarchy of palette windows in a graphical user interface displayed on a computer system, wherein the computer system includes a display, the method comprising:

displaying on the display a first palette window from the hierarchy of palette windows, wherein one or more of the palette windows in the hierarchy comprise palette items that are selectable by a user to include functionality in a program being created or modified, wherein the first palette window comprises one or more palette window selection items, wherein each of the one or more palette window selection items is operable when selected to display a different child palette window of the first palette window in the hierarchy of palette windows;

receiving user input selecting a first palette window selection item of the one or more selection items in the first palette window;

displaying a second palette window on the display in response to said receiving the user input selecting the first palette window selection item, wherein the second palette window is a child palette window of the first palette window in the hierarchy of palette windows, wherein the child palette window comprises at least one palette item; and

closing the first palette window in response to said receiving the first user input selecting the first palette window selection item.

105. The method of claim 104, wherein the palette items include icons that are selectable by the user to incorporate Graphical User Interface (GUI) elements in a GUI of the program being created or modified.

106. The method of claim 104, wherein the program being created or modified is a graphical program, and wherein the palette items include icons that are selectable by the user to include nodes in the graphical program.

107. The method of claim 104, wherein each of the palette windows in the hierarchy of palette windows comprises one or more navigation items, the method further comprising:

receiving user input selecting a navigation item of the second palette window, wherein the navigation item is operable when selected to close a currently displayed palette window and display a previously displayed palette window;

closing the second palette window in response to said user input selecting the navigation item; and

displaying the first palette window on the display in response to said user input selecting the navigation item.

108. The method of claim 107, wherein the navigation item is a back navigation item operable when selected to display a most recently previously displayed palette window in a backward direction.

109. The method of claim 104, wherein each of the palette windows in the hierarchy of palette windows comprises one or more navigation items, the method further comprising:

receiving user input selecting a navigation item of the second palette window, wherein the navigation item is operable when selected to close a currently displayed palette window and display a previously displayed palette window;

closing the second palette window in response to said user input selecting the navigation item; and

displaying a third palette window on the display in response to said user input selecting the navigation item.

110. The method of claim 109, wherein the navigation item is a forward navigation item operable when selected to display a most recently previously displayed palette window in a forward direction.

111. The method of claim 104, wherein each of the palette windows in the hierarchy of palette windows comprises one or more navigation items,

the method further comprising:

receiving user input selecting a navigation item of the one or more navigation items of a currently displayed palette window;

displaying one of a parent palette window and a child palette window in relation to the currently displayed palette window in the hierarchy of palette windows subsequent to said receiving the user input selecting the navigation item.

112. The method of claim 104, wherein one or more child palette windows in relation to a currently displayed palette window in the hierarchy of palette windows are displayable from the currently displayed palette window by user input selecting palette window selection items corresponding to the one or more child palette windows from the currently displayed palette window.

113. The method of claim 104, wherein the second palette window comprises an up navigation item, wherein the up navigation item is operable when selected to display a parent palette window of the second palette window in the hierarchy of palette windows and to close the second palette window, the method further comprising:

receiving user input selecting the up navigation item of the second palette window;

displaying the first palette window on the display in response to said user input selecting the up navigation item; and

closing the second palette window in response to said user input selecting the up navigation item.

114. The method of claim 104, wherein each of the palette windows in the hierarchy of palette windows comprises a search item, the method further comprising:

receiving user input selecting a search item of a currently displayed palette window;

closing the currently displayed palette window in response to said user input selecting the search item;

displaying a search window in response to said user input selecting the search item;

receiving user input in the search window specifying a search criteria;

identifying a new palette window in the search window in accordance with the search criteria user input;

closing the search window; and

displaying the new palette window.

115. The method of claim 114, wherein said identifying comprises:

identifying and displaying information regarding a plurality of possible palette windows in the search window in accordance with the search criteria user input; and

receiving user input selecting the new palette window from the plurality of possible palette windows.

116. The method of claim 115, wherein the plurality of possible palette windows includes palette windows from a plurality of hierarchies of palette windows.

117. A system comprising:

a memory configured to store program instructions;

an input device configured to receive user input;

a display device; and

a processor configured to read the program instructions from the memory and to execute the program instructions, wherein, in response to execution of the program instructions, the processor is operable to:

display on the display device a first palette window from a hierarchy of palette windows in a graphical user interface, wherein one or more of the palette windows in the hierarchy comprise palette items that are selectable by a user to include functionality in a program being created or modified, and wherein the first palette window includes one or more navigation items displayed on the first palette window for navigating among the hierarchy of palette windows;

receive user input from the input device selecting a navigation item displayed on the first palette window;

close the first palette window in response to said user input selecting the navigation item; and

display on the display device at least one of a parent palette window or a child palette window in relation to the first palette window in the hierarchy of palette windows after said user input selecting the navigation item.

118. The system of claim 117, wherein the palette items include icons that are selectable by the user to include functionality in the program being created or modified.

119. The system of claim 117, wherein the palette items include icons that are selectable by the user to incorporate Graphical User Interface (GUI) elements in a GUI of the program being created or modified.

120. The system of claim 117, wherein the program being created or modified is a graphical program, and wherein the palette items include icons that are selectable by the user to include nodes in the graphical program.

121. (Canceled)

122. The system of claim 117, wherein the one or more navigation items comprise a forward navigation item and a back navigation item;

wherein the back navigation item is operable when selected to display a most recently previously displayed palette window in a backward direction;

wherein the forward navigation item is operable when selected to display a most recently previously displayed palette window in a forward direction.

123. The system of claim 117, wherein the one or more navigation items comprise an up navigation item;

wherein the up navigation item is operable when selected to display a parent palette window of the first palette window, regardless of the most recently previously displayed palette window.

124. The system of claim 117,

wherein the first palette window comprises one or more palette window selection items, wherein each of the one or more palette window selection items is operable when selected to display a different child palette window of the first palette window in the hierarchy of palette windows.

125. The system of claim 117, wherein each of the palette windows in the hierarchy of palette windows comprises a search item, wherein, in response to execution of the program instructions, the processor is further operable to:

receive user input selecting a search item of a currently displayed palette window;

close the currently displayed palette window in response to said user input selecting the search item;

display a search window on the display device in response to said user input selecting the search item;

receive user input in the search window specifying a search criteria;

identify a new palette window in the search window in accordance with the search criteria user input;

close the search window; and

display the new palette window on the display device.

126. The system of claim 125, wherein, in said identifying, the processor is further operable to:

identify and display information regarding a plurality of possible palette windows in the search window in accordance with the search criteria user input; and

receive user input selecting the new palette window from the plurality of possible palette windows.

127. The system of claim 126, wherein the user input in the search window specifying a search criteria includes a search string, and wherein, in said identifying and displaying information regarding a plurality of possible palette windows in the search window in accordance with the search criteria user input, the processor is further operable to:

search for the search string in a plurality of text items related to palette windows in the hierarchy; and

display one or more located text items in the search window, wherein each of the one or more located text items includes the search string, and wherein each of the one or more located text items references one of the plurality of possible palette windows;

wherein the user input selecting the new palette window from the plurality of possible palette windows specifies one of the one or more located text items in the search window, wherein the specified located text item references the new palette window.

128. A system comprising:

a memory configured to store program instructions;

an input device configured to receive user input;

a display device; and

a processor configured to read the program instructions from the memory and to execute the program instructions, wherein, in response to execution of the program instructions, the processor is operable to:

display on the display device a currently displayed palette window from the hierarchy of palette windows, wherein one or more of the palette windows in the hierarchy comprise palette items that are selectable by a user to include functionality in a program being created or modified, and wherein the currently displayed palette window includes one or more navigation items displayed on the currently displayed palette window for navigating among the hierarchy of palette windows;

receive user input from the input device selecting a navigation item displayed on the currently displayed palette window;

close a currently displayed palette window in response to said user input selecting the navigation item; and

display a previously displayed palette window on the display device in response to said user input selecting the navigation item.

129. The system of claim 128, wherein the palette items include icons that are selectable by the user to incorporate Graphical User Interface (GUI) elements in a GUI of the program being created or modified.

130. The system of claim 128, wherein the program being created or modified is a graphical program, and wherein the palette items include icons that are selectable by the user to include nodes in the graphical program.

131. The system of claim 128, wherein the navigation item is operable when selected to close the currently displayed palette window and display a previously displayed palette window.

132. The system of claim 128, wherein the navigation item is a back navigation item operable when selected to display a most recently previously displayed palette window in a backward direction.

133. The system of claim 128, wherein the navigation item is a forward navigation item operable when selected to display a most recently previously displayed palette window in a forward direction.

134. The system of claim 128, wherein the navigation item is an up navigation item operable when selected to display a parent palette window of the first palette window, regardless of the most recently previously displayed palette window.

135. The system of claim 128,

wherein the first palette window comprises one or more palette window selection items, wherein each of the one or more palette window selection items is operable when selected to display a different child palette window of the first palette window in the hierarchy of palette windows.

136. A system comprising:

a memory configured to store program instructions;

an input device configured to receive user input;

a display device; and

a processor configured to read the program instructions from the memory and to execute the program instructions, wherein, in response to execution of the program instructions, the processor is operable to:

display on the display device a first palette window from a hierarchy of palette windows in a graphical user interface, wherein one or more of the palette windows in the hierarchy comprise palette items that are selectable by a user to include functionality in a program being created or modified, wherein the first palette window comprises one or more palette window selection items, wherein each of the one or more palette window selection items is operable when selected to display a different child palette window of the first palette window in the hierarchy of palette windows;

receive user input from the input device selecting a first palette window selection item of the one or more selection items in the first palette window;

display a second palette window on the display device in response to the user input selecting the first palette window selection item, wherein the second palette window is a child palette window of the first palette window in the hierarchy of palette windows; and

close the first palette window in response to the first user input selecting the first palette window selection item.

137. The system of claim 136, wherein the palette items include icons that are selectable by the user to incorporate Graphical User Interface (GUI) elements in a GUI of the program being created or modified.

138. The system of claim 136, wherein the program being created or modified is a graphical program, and wherein the palette items include icons that are selectable by the user to include nodes in the graphical program.

139. The system of claim 136, wherein each of the palette windows in the hierarchy of palette windows comprises a back navigation item operable when selected to display a most recently previously displayed palette window in a backward direction and to close a currently displayed palette window, wherein, in response to execution of the program instructions, the processor is further operable to:

receive user input from the input device selecting the back navigation item of the second palette window;

close the second palette window in response to said user input selecting the back navigation item; and

display the first palette window on the display device in response to said user input selecting the back navigation item.

140. The system of claim 136, wherein one or more of the palette windows in the hierarchy of palette windows comprises a forward navigation item operable when selected to display a most recently previously displayed palette window in a forward direction and to close a currently displayed palette window, wherein, in response to execution of the program instructions, the processor is further operable to:

receive user input from the input device selecting the forward navigation item of the second palette window;

close the second palette window in response to said user input selecting the forward navigation item; and

display a third palette window on the display device in response to said user input selecting the forward navigation item.

141. The system of claim 136, wherein the second palette window comprises an up navigation item, wherein the up navigation item is operable when selected to display a parent palette window of a currently displayed palette window in the hierarchy of palette windows and to close the currently displayed palette window, wherein, in response to execution of the program instructions, the processor is further operable to:

receive user input from the input device selecting the up navigation item of the second palette window;

display the first palette window on the display device in response to said user input selecting the up navigation item; and

close the second palette window in response to said user input selecting the up navigation item.

142. The system of claim 136, wherein each of the palette windows in the hierarchy of palette windows comprises a search item, wherein, in response to execution of the program instructions, the processor is further operable to:

receive user input from the input device selecting a search item of a currently displayed palette window;

close the currently displayed palette window in response to said user input selecting the search item;

display a search window on the display device in response to said user input selecting the search item;

receive user input in the search window specifying a search criteria;

identify a new palette window in the search window in accordance with the search criteria user input;

close the search window; and

display the new palette window on the display device.

143. A carrier medium comprising program instructions, wherein the program instructions are computer-executable to implement:

displaying on a display of a computer system a first palette window from a hierarchy of palette windows in a graphical user interface, wherein one or more of the palette windows in the hierarchy comprise palette items that are selectable by a user to include functionality in a program being created or modified, and wherein the first palette window includes one or more navigation items displayed on the first palette window for navigating among the hierarchy of palette windows;

receiving user input selecting a navigation item displayed on the first palette window;

closing the first palette window in response to said receiving user input selecting the navigation item; and

displaying at least one of a parent palette window or a child palette window in relation to the first palette window in the hierarchy of palette windows after said user input selecting the navigation item.

144. The carrier medium of claim 143, wherein the program being created or modified is a graphical program, wherein the palette items include icons that are selectable by the user to include nodes in the graphical program.

145. (Canceled)

146. The carrier medium of claim 143, wherein the one or more navigation items comprise a forward navigation item and a back navigation item;

wherein the back navigation item is operable when selected to display a most recently previously displayed palette window in a backward direction;

wherein the forward navigation item is operable when selected to display a most recently previously displayed palette window in a forward direction.

147. The carrier medium of claim 143, wherein the one or more navigation items comprise an up navigation item;

wherein the up navigation item is operable when selected to display a parent palette window of the first palette window, regardless of the most recently previously displayed palette window.

148. The carrier medium of claim 143, wherein each of the palette windows in the hierarchy of palette windows comprises a search item, wherein the program instructions are computer-executable to implement:

receiving user input selecting a search item of a currently displayed palette window;

displaying a search window in response to said user input selecting the search item;

receiving user input in the search window specifying a search criteria;

identifying a new palette window in the search window in accordance with the search criteria user input; and

displaying the new palette window.

149. A carrier medium comprising program instructions, wherein the program instructions are computer-executable to implement:

displaying on a display of a computer system a currently displayed palette window from a hierarchy of palette windows in a graphical user interface, wherein one or more of the palette windows in the hierarchy comprise palette items that are selectable by a user to include functionality in a program being created or modified, wherein the currently displayed palette window includes one or more navigation items displayed on the currently displayed palette window for navigating among the hierarchy of palette windows;

receiving user input selecting a navigation item displayed on the currently displayed palette window;

closing a currently displayed palette window in response to said user input selecting the navigation item; and

displaying a previously displayed palette window in response to said user input selecting the navigation item.

150. The carrier medium of claim 149, wherein the navigation item is one of a back navigation item, a forward navigation item, and an up navigation item.

151. A carrier medium comprising program instructions, wherein the program instructions are computer-executable to implement:

displaying on a display of a computer system a first palette window from a hierarchy of palette windows in a graphical user interface, wherein one or more of the palette windows in the hierarchy comprise palette items that are selectable by a user to include functionality in a program being created or modified, wherein the first palette window comprises one or more palette window selection items, wherein each of the one or more palette window selection items is operable when selected to display a different child palette window of the first palette window in the hierarchy of palette windows;

receiving user input selecting a first palette window selection item of the one or more selection items in the first palette window;

displaying a second palette window on the display in response to said receiving the user input selecting the first palette window selection item, wherein the second palette window is a child palette window of the first palette window in the hierarchy of palette windows; and

closing the first palette window in response to said receiving the first user input selecting the first palette window selection item.

152. The carrier medium of claim 151, wherein each of the palette windows in the hierarchy of palette windows comprises a search item, wherein the program instructions are further computer-executable to implement:

receiving user input selecting a search item of a currently displayed palette window;

closing the currently displayed palette window in response to said user input selecting the search item;

displaying a search window on the display in response to said user input selecting the search item;

receiving user input in the search window specifying a search criteria;

identifying a new palette window in the search window in accordance with the search criteria user input;

closing the search window; and

displaying the new palette window on the display.

153. A computer-implemented method for creating or modifying a program using a hierarchy of palette windows in a graphical user interface displayed on a computer system, the method comprising:

displaying on a display of the computer system a first palette window from the hierarchy of palette windows, wherein the first palette window from the hierarchy of palette windows comprises at least one palette window selection item that is selectable by a user, wherein the palette window selection item is associated with a second palette window from the hierarchy of palette windows, wherein the second window from the hierarchy of palette windows comprises at least one palette item that is selectable by the user, wherein the palette item is associated with functionality that can be included in the program being created or modified;

receiving user input selecting the at least one palette window selection item from the first palette window;

in response to said receiving user input selecting the at least one palette window selection item, displaying the second palette window and closing the first palette window.

154. The method of claim 153, further comprising:

receiving user input selecting the at least one palette item from the second palette window; and

in response to said receiving user input selecting the at least one palette item, including the functionality associated with the palette item in the program being created or modified.

155. The method of claim 153, wherein the at least one palette item includes an icon that is selectable by the user to incorporate a Graphical User Interface (GUI) element in a GUI of the program being created or modified.

156. The method of claim 153, wherein the program being created or modified is a graphical program, and wherein the at least one palette item includes an icon that is selectable by the user to include a node in the graphical program.

157. The method of claim 156, wherein the graphical program comprises a plurality of interconnected nodes that graphically represents functionality of the graphical program, and wherein the graphical program is executable by a computing device to perform the functionality.

158. The method of claim 153, wherein the second palette window includes a navigation item displayed on the second palette window, wherein the navigation item is associated with the first window;

the method further comprising:

receiving user input selecting the navigation item displayed on the second palette window; and

in response to said receiving user input selecting the navigation item displayed on the second palette window, displaying the first palette window and closing the second palette window.

X. EVIDENCE APPENDIX

No evidence submitted under 37 CFR §§ 1.130, 1.131 or 1.132 or otherwise entered by the Examiner is relied upon in this appeal.

XI. RELATED PROCEEDINGS APPENDIX

There are no related proceedings.